UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/731,678	12/06/2000	Sung-Hee Do	A0734/7001 (EJR)	9300

7590 06/16/2008

Edward J. Russavage
Wolf, Greenfield & Sacks, P.C.
600 Atlantic Avenue
Boston, MA 02210

EXAMINER	
VU, TUAN A	

ART UNIT	PAPER NUMBER
2193	

MAIL DATE	DELIVERY MODE
06/16/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	09/731,678	DO ET AL.	
	Examiner	Art Unit	
	Tuan A. Vu	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11 April 2008*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *96-107 and 110-113* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *96-107, 110-113* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to the Applicant's response filed 4/11/08.

As indicated in Applicant's response, claims 96, 105 have been amended. Claims 96-107, 110-113 are pending in the office action.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 96-107, 110-113 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kim et al., "Design of Software Systems Based on axiomatic Design", CIRP, 1991, pp. 243-255 (hereinafter Kim), in view of Talbott et al., USPN: 5,375,440 (hereinafter Talbott).

As per claim 96, Kim discloses method of designing a software system, comprising:

defining a set of functional requirements (FRs – Fig. 1- pg. 244) that describe what the software system is to achieve;

defining a set of design parameters, where each design parameter in the set satisfies at least one of the functional requirements (e.g. DPs – Fig. 1, pg. 244);

decomposing the set of functional requirements and design parameters to create a hierarchy of functional requirements and a hierarchy of design parameters (Fig. 2, pg. 245; chp: *Hierarchical structuring and decomposition* – pg. 246), wherein at least one functional requirement of the set of functional requirements is a parent functional requirement at a first level in the hierarchy of functional requirements and is decomposed into at least two child

functional requirements at a second level in the hierarchy that is below the first level, and wherein the at least two child functional requirements collectively accomplish the parent functional requirement (e.g. FR1 → FR11, FR12 – Fig. 2;, *step 1: FRs → DPs*, right column, pg. 246);

defining a design matrix (e.g. *design matrix* – eq. (11), pg. 250; eq. (12), pg. 251) that maps each design parameter in the hierarchy of design parameters to the at least one functional requirement in the hierarchy of functional requirements that the respective design parameter satisfies (*step: 1 → step 6, 7*, pg. 246-248; Fig. 4); and

using the design matrix to define the functional requirements (FRs) composing the software system (e.g. eq. *(7)*, pg. 246, R col, bottom; *Hierarchical structuring and decomposition* – pg. 246 – Note: *FR1, FR2 ... FR11, FR12* methods to implement the needs of library software system reads on software object modules – see Fig. 7-8, pg. 249; *software modules ... best software package* - pg. 253, L col. eqs. 12-13 reads on software object represented by one FR) wherein at least one FR represents a software object (e.g. *Each FR has one module of software* – pg. 248, R col. middle; *software system ... FR hierarchy* - pg. 246, L col. bottom; *FR1= Generate the DB ..., FR2 = Provide a list of references* - R col. pg. 246; *modules* – Fig. 5, pg 248) of the software system;

wherein at least one design parameter in the hierarchy of design parameters represents an *attribute* of the software object (e.g. Fig. 1; *in terms of process variables ... process domains are in form of subroutines ... subsubroutines* --top L col., pg. 244 Note: DPs to represent *shape, structure, loads, geometries, Cross-sectional shape, number of ribs* --see *DP1, DP2* – pg. 250 L

col.; *DP11, DP12, DP21, DP22* - pg. 250 R col. – reads on attribute associated with a DP satisfying at least one software object represented by a FR of the matrix)

But Kim does not explicitly disclose FR-representing software objects or modules are object-oriented structure. The concept of *object-oriented* methodologies and CASE tool have been considered known when Kim introduced the axiomatic approach (see *Object-oriented Software* - Introduction, L column, pg. 243; CASE – pg. 248; *hierarchy ...divide and conquer* – pg. 252, L col.; Fig 5, pg. 248) in that Kim teaches decomposition of FR into software parent/child modules (e.g. Fig. 5, pg. 248; CASE – pg. 248; *child, parent* - see pg. 249, L column, i.e. suggestive of object hierarchies) and matching of database-stored legacy of DPs or FRs to obtain libraries of software packages or pre-existing modules that satisfy a axiomatic equation, as in a vertical integration process being dependent upon other existing modules organized as top-down layers based on hierarchy of FRs (e.g. pg. 253, L bottom to R column - Note: *database, hierarchy... library ... software package* - pg. 253, R column -- indicative of software package in layers and persisted in package for reusability), hence the concept of reusable object in development. Similar to the modules in Kim (see Kim: Fig. 5, pg. 248) in view of Kim's design matrix, Talbott discloses development framework using CASE tool (see Talbott: Fig. 2; col. 10, lines 20-30) to implement a specific industrial application with objects of a domain organized in an requirements structured in a matrix correlating requirements and attributes of object organized in hierarchies, propagation in levels of objects hierarchy using said *matrix* to readjust to impact incurred to attributes for each levels (e.g. Talbott: Fig. 6; *higher levels lowest ... levels ... propagate ...limits and goals downwardly ... upwardly* – col. 6 lines 27-56; col. 20, lines 5-39; *Action/Object ... propagation* –col. 13 line 6 to col. 14, line 12), hence

suggesting object-oriented hierarchical levels with properties or functional inheritance from top to bottom levels. Based on the well-known concept of parent-child hierarchy and available Object-oriented methodologies in software development (e.g. CASE Tools software development; *Object-oriented Software* - Introduction, L column, pg. 243) as suggested in Kim at the time the invention was made, along property/action relationship among Parent-Child-oriented objects as shown by Talbott, it is recognized that object oriented in CASE tools with hierarchizing of parent/child software modules as in Talbott and Kim to implement complex software system (or industrial process) is reminiscent of well-known object-oriented framework where objects are reusable and equipped with OO language such as inheritance whereby attributes/action can reused from parent to child levels, so to alleviate code creation, as set forth above; that is, a framework typically accommodating object-oriented language with CASE in conjunction with Kim's purport to apply *reusable* software set forth in Kim's (see *develop reliable and reusable software* – see Abstract, pg. 243). It would be obvious for one skill in the art at the time the invention was made to implement the modules associated with each FRs as intended by Kim, so that these modules being stored in existing libraries or legacy database be reuse object-oriented packages or modules as exemplified in Talbott (see *reusable ... minimizing cost* – col. 15, lines 24-45), because the creation of OO instances as they are retrieved from reuse can support the non-dependency of module being fetched in Kim's process of integration as purported in the axiomatic matching as set forth above, thus alleviating source code recreating resources usage via reuse of pre-stored objects (see Talbott, col. 15), such that has been widely practiced in CASE Tools as mentioned above, and effectively applying thereby object-oriented

language inheritance to alleviate the propagation of requirements and readjusting of parameter in a design matrix as purported in Talbott, without extensive recreation of new code.

As per claim 97, Kim discloses that at least one element of the design matrix and the at least one design parameter represents an operation performed by the software object (see *FR*x, *DP*x - equations *7-12*, pp. 246-248, 250-251; *FR1= Generate the DB ..., FR2 = Provide a list of references* - R col. pg. 246; *modules* – Fig. 5, pg 248).

As per claim 98, Kim discloses that wherein the act of defining the set of define parameters further comprises determining the set of design parameters by mapping the set of functional requirements into a physical implementation domain (e.g. *physical domain* – pg. 251, R column).

As per claims 99-100, Kim discloses an act of determining if the design matrix is decoupled (eq. 11, pg. 250); and is not decoupled, manipulating the design matrix into lower triangular form (e.g. pg. 249, L column; eq. 11, pg. 250).

As per claim 101, Kim discloses wherein the at least one functional requirement that represents a software object includes at least two functional requirements, and wherein a first of the at least two functional requirements represents a first software object and a second of the at least two functional requirements represents a second software object (e.g. Fig. 2, 4, 5, pg. 245, 247, 248, respectively).

As per claim 102, Kim discloses defining a relationship between the first software object and the second software object using a junction (e.g. pg. 249, L column, Fig. 7).

As per claim 103, Kim discloses defining a third software object by combining the first software object and the second software object according to a type of the junction (e.g. *Summing Junction* - Fig. 7, pg. 249).

As per claim 104, Kim discloses wherein the type of the junction is one of: a summation junction; a control junction', or a feedback junction (e.g. pg. 249, L column; Fig. 7).

As per claim 105, Kim discloses one computer readable medium encoded with instructions that, when executed on a computer system, perform a method of allowing a user (e.g. *framework for software design* – pg. 243, R col.) to define a software system, the method comprising allowing the user to:

define a set of functional requirements that describe what the software system is to achieve;

define a set of design parameters, where each design parameter in the set satisfies at least one of the functional requirements;

decompose the set of functional requirements and design parameters to create a hierarchy of functional requirements and a hierarchy of design parameters, wherein at least one functional requirement of the set of functional requirements is a parent functional requirement at a first level in the hierarchy of functional requirements and is capable of being decomposed into at least two child functional requirements at a second level in the hierarchy that is below the first level, and wherein the at least two child functional requirements collectively accomplish the parent functional requirement;

define a design matrix that maps each design parameter in the hierarchy of design parameters to the at least one functional requirement in the hierarchy of functional requirements that the respective design parameter satisfies; and

using the design matrix to define an *object-oriented* structure of the software system (by virtue of obviousness rationale set forth in claim 96), wherein at least one functional requirement in the hierarchy of functional requirements represents a software object of the software system, and wherein at least one design parameter in the hierarchy of design parameters represents an attribute of the software object;

all of which limitations having been addressed respectively in claim 96.

As per claims 106-107, and 110-113, these claims correspond to the subject matter of claims 97-98, and 101-104, respectively; hence are rejected using the rationale set forth therein, correspondingly.

Response to Arguments

4. Applicant's arguments filed 4/11/08 have been fully considered but they are not persuasive. Following are the Examiner's observation in regard thereto.

USC 35 § 103 Rejection:

(A) Applicants have submitted that neither Kim nor Talbott discloses 'using a design matrix to define an object-oriented … represents an attribute of the software object' because, as facts proffered by the Office Action, software discussed in Kim for cataloguing books includes functional requirements such as generating a call number and providing a list of references; i.e. nowhere in Kim are there any FRs that can implemented in Object-oriented system, any FR to represent a software object in a OO software system (Appl. Rmrks, pg. 7, middle and bottom).

The rationale for obviousness has been established based on teachings by Kim, like hierarchy of parent/child modules, well-established methodologies such as CASE and object-oriented development approaches, design matrix mapping FR to DP; by Talbott using matrix to correlate what a change in a FR would have to be impacted in a attribute similar to the DP/FR matrix by Kim, the properties/action and the parent/child relationship in terms of properties propagation in Talbott, and level of one of ordinary skill in the art when presented with all the above pieces of teaching thus recognized, including known practices at the time the invention was made. The rationale as set forth in the rejection is to render obvious the implementation of Kim's modules when these are hierarchized as FRs to be matched with the corresponding DPs of the design matrix such that this implementation uses object-oriented approach, with object containing action or attributes reminiscent of OO language with well-known properties like inheritance, reusability etc.; and in presenting this rationale, the Rejection has laid out suggestion, existing well-known technologies, teachings from both Kim and Talbott, in order to yield a combination with clear explanation as to how as combined the presented teachings would benefit the endeavor like code reuse by Kim to address complex software process in light of Talbott, based on level of the one of ordinary skill in the art as mentioned above. The Examples about the library cataloguing software is to illustrate that any FR is to represent a task, an activity or a software function, a component among others in the modular hierarchy represented by the FRs in Kim's matrix; whereas the mainstay of the rejection is relying of both Talbott and Kim, as laid out above. In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on

combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

(B) Applicants have submitted that the declaration by Dr. Kim has made clear that Kim does not disclose or suggest implementing FR as OO class or package (Appl. Rmrks pg. 8, top). In reply, the declaration filed 7/14/2006 has been deemed addressed in a previous action as insufficient; because the presentation of facts to overcome the rejection appeared to be just remark or allegation without convincing evidence corroborating thereto.

(C) Applicants have submitted that amended, Kim fails to disclose 'design parameters represents an attribute of the software object'; this remark (Appl. Rmrks pg. 8, middle) founded upon a change in a claim language would be considered moot in view of the adjusted grounds of rejection. Moreover, Applicant's submission that Talbott fails to cure the infirmities of Kim is deemed mere allegation for patentability without proper prima facie case in order to establish how the teachings in Talbott, in view of well-known concepts, fail to support the Examiner's legal conclusion, based on what is suggested or evidenced in Kim, that OO type of language can be applied in implementing Kim's modules to enhance Kim's purport to address complex software development via a efficient reuse and resource-alleviating manner. The argument is not persuasive.

(D) Applicants have submitted that while CASE tools were known, Applicants are unaware of any system that is for defining an object-oriented structure in a manner as recited in claims 96 and 105 (Appl. Rmrks pg. 9, top) because neither Kim nor Talbott taken alone or in combination fulfill such system. The argument is referred back to sections A to C above.

The claims will stand rejected as set forth in the Office Action.

Conclusion

5. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (571) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lewis Bullock can be reached on (571)272-3759.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence - please consult Examiner before using) or 571-273-8300 (for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Tuan A Vu/

Primary Examiner, Art Unit 2193

June 13, 2008